POPULAR

Computing WEEKLY

30p

Play Planet Ruler on ZX81



BBC graphics

Music on Vic

Win a 16K RAM pack

APPLE II **COMMODORE**

ATLANTIC
MICRO SYSTEMS

72 Honor Oak Park, London SE23 1DY
Telephone 01-699 2202
Telex No 896691 TLXIR G ATLANTIC

Atlantic VIC-20 CASH Prices

Vic-20 Computer	£155.95
Vic Cassette	£[illegible]

SPECIAL OFFERS

Oki Microline 80 printer (RRP £299.00)
£260.00 + VAT

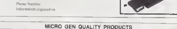
Ricoh 1600S Daisywheel (RRP £1 600.00)
Printer £1 295.00 + VAT

For further information give us a call and ask for our trade desk. If you prefer to write please complete the form below

Name
Address
Phone Number
Information required on

Buy our **£475** + VAT Daisy Wheel Printer for your computer and you have an 'Electronic Typewriter' absolutely FREE





MICRO GEN QUALITY PRODUCTS

ZX81 A/D CONVERTER BOARD

Th[illegible] can be used [illegible] voltage, temperature, light intensity etc.
The board fits in between the RAM pack and the ZX81 [illegible] make the connection and [illegible] improves the stability of the RAM pack.
Price [illegible] £18.50

JOYSTICKS FOR THE ZX81, only £[illegible].50 each

- [illegible] an expensive keyboard
- [illegible]
- [illegible]
- Extends the capability of the ZX81 – imagine the tremendous variety of games and applications and now become possible
- Detailed supplied on how to use the joysticks in your own programs

Please note that you cannot connect conventional analogue joysticks directly to the digital input ports found on most I/O boards, an A/D converter such as ours is required.

A free copy of ZX AMAZE plus any one of the games listed below when ordering a Joystick and an A/D board.

Programs available

ZX SPACE INVADERS You've tried the rest, now try the BEST
[illegible] many features including an ever increasing rate [illegible] (they'll get you in the end) only £3.95

ZX BREAKOUT Quite simply the best breakout on the market
[illegible] angles (you won't find this one easy) now only £3.95

ZX NEW YORK A very addictive game
[illegible] way out of trouble, otherwise you are bound to crash. Generates a different pattern, for a different game each time you play. On the reverse of the cassette is ZX REFLEX, find out how fast you really are now only £3.95

ZX CHESS The original and still the best ★ Graphic display of chessboard ★ Six levels of play ★ Displays record of your moves and the computer's ★ Board can be set up in any position ★ Has ability to change sides or level in mid-game ★ PLUS ★ CHESS CLOCK on reverse side – records time taken by each player ★ Resettable function ★ Single key entry **now only £6.50**

All our games are written in machine code and can be used with joysticks or keyboard (except chess, keyboard only). Each on cassette with library case.
To allow you to prove to yourself that our products are second to none, MICRO GEN offers the following terms to our customers

14 DAYS FREE APPROVAL ON ALL PRODUCTS

All money will be refunded if goods are returned in good condition within 14 days of despatch.
If you write a program which is exceptional, please submit it to us. We will offer a royalty [illegible]. Cheques and PO's payable to MICRO GEN, Dept [illegible], 24 Agar Crescent, Bracknell, Berks.

POPULAR Computing WEEKLY

The Team

Editor
Duncan Scot

Production consultants
Myles Hewitt
Eric Robbie

Editorial Secretary
Fiona McCormick

Advertisement Manager
David Lake [01-839 2849]

Advertisement Executive
Peter Chandler [01 839 1881]

Publishing Director
Nick Hampshire

Popular Computing Weekly,
Hobhouse Court
19 Whitcomb Street,
London WC2 7HF
01-839 6835 and 01-839 1855

Published by Sunshine Publications Ltd
Hobhouse Court 19 Whitcomb Street
London WC2 7HF

Typesetting origination and printing by
Chesham Press Chesham Bucks

Distributed by S M Distributors
London SW9 01 274 7421 Telex 261643

Subscriptions
You can have *Popular Computing Weekly* sent to your home the subscription will be £19.95 per year for addresses in the UK.

How to submit articles
Articles which are submitted for publication should not be more than 2,000 words long.

If you are writing for a particular section, such as Programming, you should try to cover the ground in about 1,000 words.

All submissions should be typed and a double space should be left between lines.

Programs should, whenever possible, be computer printed.

We cannot guarantee to return every submitted article. *Popular Computing Weekly* can accept no responsibility for any errors in programs we publish, although we will always try our best to make sure programs work.

This Week


Cover [illegible]

Editorial

In 18th Century London prices of goods in shops were rarely fixed. Customers bartered and haggled for everything.

Near the end of the century Messrs Flint and Palmer opened a shop on London Bridge. Every item had a set price and the shop assistants refused to argue.

At first, it is reported, customers hardly knew how to react but the new system became very popular.

The booming personal computer market could see a change back to bartering. It is certainly worth your while to shop around. When Clive Sinclair launched his 16K memory pack the price was £49.95. The cheapest of the 16K packs now on the market is £35.

Over the next few months we can expect to see many new companies enter the market and then prices will start to tumble. In the meantime Vic 20 owners will realise that theirs is a buyers' market.

If they do not like the prices all they have to do is argue. Or wait.

Next Week


Have you got what it takes to be a Hell driver? Find out in next week's fantastic fast moving listing


ZX81
SOFTWARE
High Resolution
special techniques
...quality
£34.95
Music
Nightmare Park
Space Invaders
Life
No hardware modifications whatsoever
S.A.E. for catalogue
All programs supplied on cassette with instructions
MACRONICS
West Midlands
When adult education becomes child's play!
BIG SAVINGS ON
ATARI 400
SPECIAL PRICE ONLY
£299.95
INC VAT
MAIN DEALERS FOR
ACORN - VIC 20
B.B.C. - ATARI
AND SOFTWARE SPECIALISTS
It's here at last!
B.B.C Microcomputer
Games Pack 1
ACORN ATOM Packages
for further information send large S.A.E.
Computers for All
72 NORTH STREET, ROMFORD, ESSEX. TEL 0708 60725

Will Clive sell out?

By the end of the year Sinclair's company may not be quite such a personal affair as it is at present.

Clive intends to offer 10 per cent of the shares in his fast growing enterprise for sale to investors on the Stock Exchange. Currently he owns 85 per cent of the shares of Sinclair Research, with a friend of his accounting for the other 5 per cent of the share capital.

Clive has hired Rothschild the merchant bank — which by the way handled the controversial sale of the state owned Amersham company to private investors — to take care of his proposed debut on the Stock Exchange.

At present Rothschild is preparing a report on the matter for Clive, after which he will make a final decision about whether to go ahead.

A spokesman for Sinclair said, "there's likely to be a share placement of about 10 per cent of the company to fund research and development projects including an electric vehicle without drawing off too much money from the computer side of the business."

Chess machines give tough opposition

Leading chess computers came up against stiff opposition from humans at the first Silica Chess Computer Symposium sponsored by Kent based games specialist the Silica Shop and staged at London's Imperial College.

Machine	Won	Drew	Lost	Grading (BCF)
Chess Champion Mk V	4	8	22	124 + −18
Champion Sensory Challenger	8	2	11	113 + −22
Great Game Machine (Morphy)	5	2	12	122 +/−30

Manufacturer's claims range from BCF 148 to above 160.

How the chess computers fared against the humans

The machines acquitted themselves respectably but their final ratings were below the manufacturers' claims according to the organiser John White.

"As the sophistication of chess computers has increased, it's become increasingly difficult to find out just how strong they really are," said White.

"The manufacturers used their estimations on contests between computers or limited trials against humans, and it's uncertain how reliable their claims are. The symposium was designed to answer these and related questions."

The machines tested were those which the organisers believe to be the strongest currently available — the Great Game Machine from Applied Concepts (with Morphy, Grunfeld and Capablanca chess cartridges), the Champion Sensory Challenger from Fidelity, and the Chess Champion Mk V from Sci-Sys.

Time settings were made for an average play rate of 2½ minutes per move, with adjudication after 60 moves by the international master Bob Wade.

Two major teams played the machines — one raised from London and one from Berkshire. A third team was raised on the spot from volunteers from these two teams and from spectators.

"All the players were surprised and impressed by the strength of play of the machines, which drew many favourable comments," said White. "The machines will give hard games against most experienced club players and will massacre the inexperienced player."

The table shows the main results of the contest. A full account of the symposium with detailed game scores is available from Silica Shop, 1-4 The Mews, Hatherley Road, Sidcup, Kent DA14 4DX.

DCP announce a ZX81 speech pack

The first plug-in speech pack for the ZX81 has been announced by DCP Microdevelopments of Norwich.

Described by DCP as a fully enclosed compact unit which plugs directly into the ZX81, the new pack offers an in-built speaker, volume control and extension earphone/speaker socket, and takes its power from the ZX81.

A word-pack ROM is fitted as standard to give numbers from zero to over 1 million, the complete alphabet and a number of other words. Up to three more 8K word pack ROMs can be added to provide a 300 to 400 word vocabulary.

DCP managing director David Palmer said: "We expect most of our early sales to come from users who want to use speech in games, but the speech is also suitable for more serious applications such as aiding blind users. The speech output does not sound at all 'mechanical' — in fact it is very realistic."

The DCP Speech Pack is available at £49.95 inc VAT, while the word-pack ROMs cost £15.95 inc VAT. They can be obtained direct from the manufacturer.

The address to write to is DCP Microdevelopments, 2 Station Close, Longwood, Norwich NR13 4AX.

Commodore show is biggest yet

Commodore will hold its annual exhibition at the Cunard Hotel in Hammersmith, London on June 3-5.

Ponderously entitled The Third International Commodore Computer Show — in the old days it was known to one and all simply as the Pet show — this year's event will be (surprise, surprise) "the biggest and best Commodore Show yet" according to the organisers.

Commodore expects more than 20,000 visitors and says that more than 100 exhibitors will display a wide range of hardware and software products to cover applications including specialist business uses, education and communications.

Among the products to be shown by Commodore itself is the forthcoming Ultimax small computer (see our news story in last week's issue of *Popular Computing Weekly*). Hobbyists beware, however: Commodore admits that the emphasis of the show will be on business, because this is where the company sees most of its sales coming from.

A number of specialist seminars will run throughout the show, and guest speakers will include Jim Butterfield, the internationally known authority on the Commodore Pet computer.

Y ZXers should cum to Brum

Microscene Brum 82 is the name of a three day exhibition concentrating on the Sinclair ZX computers, to be held on Saturday September 11 at the Bingley Hall in Birmingham.

The show is being organised by Eric Deeson, well known for his activities with the ZX computer.

Further details are available from Microscene, 6 Baltonsborough Road, Harborne, Birmingham 17.

You get £50 for starting up a User Group!

***Pete Gerrard** tells you how to go about setting up a Commodore User Group*

It didn't take too long after the first microcomputers started appearing for the very first User Groups in this country to begin springing up.

These were usually started up by enthusiastic amateurs who wanted to pool their resources in order to further their own individual knowledge concerning whatever particular micro they owned.

Given time these User Groups got on to a much firmer footing and nowadays the most successful of these is probably the Independent PET Users Group or as they recently renamed themselves the Independent Commodore Products Users Group.

ICPUG (as we shall call them from now on) have their own chairman, secretary, treasurer and so on and regularly hold committee meetings in order to determine their future path.

It soon became apparent that running a nationwide set of groups was more than just one body of people could handle and so ICPUG split up into a number of regional sub-groups. Today there are 20 of these regionals all coming under the ICPUG umbrella of organisation.

As a point of information the contact for any queries, offers of help etc the person to contact is Mrs Ell Pamphlett, at 7 Lower Green, Tewin, Welwyn, Hertfordshire (Tel Welwyn 7325).

One name for all

The reason for the renaming of the group should be fairly obvious. With the appearance of the Vic 20 Commodore introduced another computer on to the scene and so the label "Commodore Products" became more logical than "PET". This also allows the User Groups to encompass any future products that appear from the Commodore stable.

It is the Vic side of ICPUG that we'd like to concentrate on here.

What is the purpose of joining, organising, starting or indeed having anything to do with a Vic Users Group?

Mike Todd [illegible]. He is looking for would-be User Groups

User Groups exist for many purposes, not least of which is the dissemination of information gained from many sources.

Secondly User Groups can organise seminars, attend exhibitions and so on, again something that one person alone cannot do.

Thirdly if as ICPUG have you can establish a good working relationship with the manufacturer whose products you're using, you stand a very good chance of being kept abreast of all the latest information regarding that company.

Finally it is a chance to meet with fellow enthusiasts and find out for yourself what others have already discovered.

So what are ICPUG doing in terms of the Vic and how do they hope to proceed in the months to come?

The main man behind the current set-up is Mike Todd, whose name you may have encountered in *Vic Computing* as the writer of many interesting articles on the Vic 20. In his spare time Mike works for BBC Radio and could arguably be described as the most knowledgeable man in the UK on the Vic at present.

It is his role to get the Vic machine halting and with that in mind he is looking for people around the country to begin setting up Vic User Groups.

As stated earlier there are 20 PET groups at the moment. If you contact Mrs Pamphlett at the address given earlier she'll be able to tell you where they all are.

Initially ICPUG would like to see Vic people joining existing PET groups. If the demand is sufficient these can later disassociate themselves from the PET side of the group, but of course keep in with ICPUG and retain all the benefits which that entails.

The kind of thing that ICPUG are looking for from people keen on setting up a User Group is (a) people who can bring in their own software and more important (b) people who are willing to convert the vast library of PET programs that already exist to work on the Vic.

If enthusiasm isn't enough to convince you ICPUG are willing to offer up to £50 to cover the initial expenses of setting up a group. This will be necessary to cover mail-outs and the like.

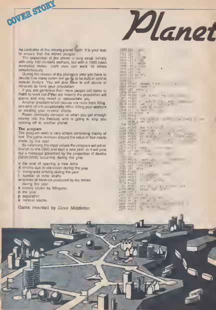
As well as this ICPUG are in the position to receive a vast amount of information from Commodore, both technical and promotional, and are more than willing to disseminate that to people who go about organising a Vic group.

Maybe a Vic group

If enough get together on this it is possible that at some future date an IVUG might appear with its own committee and so on. However time alone will tell on this one.

A starting point would be to go to the ICPUG stand at the IPC Computer Fair. This is stand number 761 at the end of a whole row of user group stands where you'll find people willing to offer you help and advice on how to start the whole ball rolling.

> Write to Club Reports, *Popular Computing Weekly*, Hobhouse Court, 19 Whitcomb Street, London WC2 7HF with details of successes you have had with your club, with ideas for helping clubs along and with any news of special meetings. We look forward to hearing from you.



As controller of the mining planet Raith it is your task to ensure that the mines prosper.

The population of the planet is very small initially with only 100 resident workers, but with a 1000 robot controlled mines each man can work 10 mines simultaneously.

During the course of the planetary year you have to decide how many mines are going to be built or sold to outside traders. You will also have to sell stocks of minerals to feed your population.

If you are generous then more people will come to Raith to work but if you are miserly the population will starve and may revolt or assassinate you.

Another problem which occurs are raids from Klingons who attack occasionally either killing your workers or stealing your mineral stocks.

Power obviously corrupts so when you get enough money into the treasury who is going to stop you running off to another planet!

The program

The program itself is very simple consisting mainly of text. The game revolves around the value of four inputs made by the user.

By balancing the input values the program will either branch to line 2000 and start a new year or it will print out a message governed by the proportion of deaths (3220-3250) occurring during the year.

v the cost of opening a new mine
d deaths due to starvation during the year
I immigrants arriving during the year
I number of mine shafts
m tonnes of minerals produced by the mines during the year
k tonnes stolen by Klingons.
n the year
p population
a mineral stocks

Game invented by *Dave Middleton*

Ruler



Reviews
software

Snake

Deltasoft Osterfeldstrasse 78d, D2000 Hamburg 54 Germany
ZX81 16K, cassette price £4.50

This is one of a number of ZX games cassettes from this active supplier who publishes many items of serious software as well as bringing American material to Europe.

Snake is not new — you may well know it already. However this is the first published ZX version I've seen. It is mainly in machine code.

The snake parades around its — er — pen under your control. Every so often a mouse appears (a square of four graphic As). The snake must catch the mouse before the latter disappears again. A caught mouse increases the snake's length by one block.

As it parades, the snake must not touch itself or the boundaries of the pen. The game also stops if you can reach a snake length of 100 units.

I couldn't succeed in that, despite my choice of ultra-slow snake speed and ultra-slow mouse retreat. Such a choice, determined by fear of ulcers, makes a very torpid but still most compelling game.

At the higher speeds you must be a maestro to get anywhere.

Once you have got the hang of this game — OK. However I found some early problems which shouldn't have been there. Thus numerical inputs are not mug-trapped and the ranges are not explained, while the introductory layout is poor.

(There are also a few seconds of FAST as soon as the program has loaded — unwise programming, because it makes you think the loading has crashed.)

The major early problem, and the others quite unnecessary, is that you have to spend quite a while sorting out how to control the snake. The minimal paper documentation talks of using four direction keys — but they are not the four other folk use. Indeed all the keys I tried work.

Maybe this is to allow you to choose your own on the basis of comfort or which hand you use; maybe it is to make the game harder. Whatever the reason, I found this a waste of time.

Snake loads readily (it is recorded at high level, so beware) and is fair value for money, bearing in mind its overseas origin. It takes up about 8K.

Summary
A compelling game — but some programming deficiencies. **KJ**

Tunesmith

The Vic Centre, 154 Victoria Road, London W5.
Cassette price £6.95.

Essentially this is a simple music composing program allowing the writing, editing and saving of fairly straightforward tunes.

The number of notes you can enter obviously varies with the amount of RAM you have on board. On a Vic with 3K expansion (the minimum configuration) you have 50 notes.

The program is basically menu-driven, ie on first running it you're presented with a series of options from which to proceed. The first course of action is to enter program mode and type in your magnum opus.

This is done straightforwardly enough by answering the question How many notes? and then just typing them in one after the other.

To each note you have an optional accompanying drum beat. This is followed by a request for how fast you want the tune to be played.

Next, how many times you want each note to be played, and finally whether you want them played singly or continuously.

Apart from the options for playing, loading or saving your work of art, and the option to quit the program, there is the sixth option — editing.

Summary
This is an easy enough package to use. You won't end up producing something the London Symphony Orchestra would be proud of, but everyone has to start somewhere. **PG**

Basic programming on the BBC Microcomputer

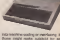
By Neil and Pat Cryer, published by Prentice-Hall, 195 pages paperback, price £5.95

'Approved by Acorn Computer' shouts the cover of this book. And so it should be, because it goes a long way towards making up for the absence of a manual for the BBC machine.

Neil and Pat Cryer are experienced programmers, and with the help of their two children and Acorn Computers they cover the BBC Micro well — extremely well.

But this is no book for the lone novice. In fact the BBC computer can't really be considered as a beginner's machine. Serious programming for it is very complex and requires a good deal of computer knowledge.

The book is comprehensive — it goes from switching on through Basic and BBC Basic to special facilities. On the way it deals with colour programming, graphics, functions, string handling, user defined functions, procedures, file handling, programmable characters and sound. It does not go



into machine coding or interfacing, but those might make subjects for later volumes.

Because the material is of necessity cramped, the book is not always easy reading. But there are plenty of programs, exercises and ideas, and the exercises are followed by helpful discussion. The book closes with a not very useful summary of BBC Basic instructions and a detailed index.

Summary
The book is sometimes superficial and can be hard-going in places. But it still offers outstanding value. **ED**

Reviews
hardware

Vic Super Expander

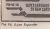
Available from any Commodore Vic distributor
Price £34.95 inc. VAT

There are so many Vic add-ons on the market with more appearing daily that it is difficult for a newcomer to computing to decide which one to buy. It makes a nice change therefore to find a product which is extremely useful and not that expensive!

The Vic Super Expander plugs into the back of the Vic and on power-up you notice the first visible sign of the cartridge in operation — you now have 6519 bytes to play with.

The next feature you'll notice is that the Vic's function keys can be used meaningfully! For instance pressing function key 1 produces the word GRAPHIC on the screen and all the other function keys have similar commands associated with them.

More important than this, however, is the use of the Super Expander to



The Vic Super Expander

reassign the commands. I've now got into the habit of assigning function key 1 to be the keyword RUN plus a carriage return. Thus pressing key 1 runs your program at a single key stroke.

The limit to the amount of code you can assign to a single key appears to be four lines of the Vic screen, or 88 characters (less the five needed to set the command in operation).

There are nine graphic commands in total together with a further command for controlling sound and seven functions which enable you, among other things, to read the position of a game paddle, a joystick or a light pen.

The most frequently used commands will probably be DRAW, CIRCLE and PAINT which when using the high resolution plotting mode of 1024 by 1024 enable you to produce very easily the most incredible patterns and designs, and are very useful in illustrating a point in an educational program for example.

Summary
This is a very useful package which is highly recommended. **PG**

DCP Control System

DCP Microdevelopments, 2 Station Close, Lingwood, Norwich. Prices: Basic P-Pack – £37.95, A-Pack – £19.95, C-Pack – £29.95. All include VAT and postage.

DCP has developed a control system for the ZX80 or ZX81. It consists of 5K of RAM for Basic or machine code and a single input/output port. The port can be used to control many devices and two extra packs may be used with it to enhance the type of devices that can be controlled.

The P-Pack: the system is based on this pack which contains 4K of extra RAM to add to the internal 1K of the basic computer. This consists of two 6116 dynamic RAM chips.

The port exists in the space where the sixth K of RAM would exist. This means that the system cannot be expanded as it is without getting inside the pack, but as it is intended to only be a control system in a school or situation it should be adequate.

The port appears on twenty 0.1mm spaced molex pins at the back of the pack, eight of the pins being data lines and the other two being +5 volts and 0 volts. Only the output of the port, which is located at 22500, is latched.

The sockets to fit the pins can be obtained quite cheaply from electronic components stores. As the port is memory mapped into the RAM space it can be Peeked and Poked like a piece of RAM, making it easy to use in machine code or Basic.

The A Pack: the A Pack is used with the P Pack to provide a single analogue to digital and digital to analogue converter. The P-Pack must be used to make the connection to the computer.

The pack can translate a voltage into a digital number for the port to put on its input data lines. A full eight bit resolution of a voltage between 0 and 2.55 volts can be obtained by Peeking the port. The number output by the port is converted into a voltage within the same range.


The DCP C-Pack

The C-Pack: this pack contains eight reed relays controlled by the state of the bits from the output of the P Pack. These relays operate one contact which can switch up to 12 volts at a maximum current of 1 amp. The contact is closed by the appropriate bit going to binary 1.

The input of the pack is directly connected to the input of the P Pack, as it requires a voltage of 2.5 volts to operate it. If nothing is connected, however, eight resistors connect the inputs to 0 volts, thus the port result will give 0 if nothing is connected.

I have two reservations about the C-Pack. One is that the reed relays operate to a binary 1. This means that when the ZX80 or ZX81 is switched on all the relays will operate, closing the switches.

As the switches will more than likely be used to control various devices, they will all operate on switching on the computer. Precautions should be taken therefore so that the devices to be controlled cannot be used before the program starts.

Summary
The other observation is that the inputs to the C-Pack are not isolated from the port, so that any voltage above +5 volts could burn out not only the C-Pack but the P Pack as well.

These packs will be very useful for demonstrating the use of a computer in a school or college. They would also be useful in a simple control application. In a more complicated one they have the limit of only one port. **DH**

What happens when you peek inside & poke it?

Nick Hampshire explains exactly what those two often misunderstood terms mean

The Basic commands Peek and Poke are often regarded by the uninitiated as very mysterious. But they should not remain in the least mysterious because they are very useful commands in a wide range of applications.

The Peek function, as its name implies, allows the programmer to look at the contents of a specified memory location. The Poke statement allows the programmer to place a value in a specified memory location.

Before examining some of the applications, let's look at how Peek and Poke are used. Both commands have fixed formats involving two variables — the memory address, and the contents of that address.

The address variable, which in the following examples has the variable name A, must ed an integer value in the range 0 to 65 535. The contents variable, called C in the examples, is also an integer variable and has the numeric range from 0 to 255. The formats of the two commands are as follows:

```
LET C = PEEK A
```

which loads the contents of memory address A into variable C, and

```
POKE A, C
```

which places the value in variable C into memory location A.

On the ZX81 the monitor — the machine code program stored on a ROM within the machine which allows one to write programs in Basic and use the machine's keyboard, display and so on — is located in the bottom 8192 locations, the bottom 8K of memory. The RAM memory which is used to store the user programs and variables starts at location 16384 and if the machine is expanded can go up to 32767.

The contents of both types of memory can be looked at with the Peek function but only the contents of RAM memory can ed changed with the Poke statement.

Applications for Peek and Poke

If you have more than 3.5K of memory in your ZX81 then the Peek and Poke commands can ed used with great effect to generate video displays. The video display on such a machine is stored as a file of 24 character strings each 32 characters long and terminated in a Newline character. This 793 byte long display file is stored in RAM memory at a starting address contained in locations 16396 and 16397 and an end address stored in 16400 and 16401.

PEEK means to look at a given address; POKE means to add to or change it

The following little program will print out all the Ascii character codes of the current contents of each display location on the screen.

```
10 LET S = PEEK 16396+256*PEEK 16397
20 LET E = (PEEK 16400+256*PEEK 16401) - 1
30 LET L = E - S
40 DIM X(L)
50 FOR Q = S TO E
60 LET X(Q-S) = PEEK (Q)
70 NEXT Q
80 FOR Q = 1 TO L
90 PRINT X(Q)
100 NEXT Q
```

Just as the above program will show the Ascii codes of each character stored in the display file we can use Poke to put characters on the display. Try the following program:

```
10 LET S = PEEK 16396+256*PEEK 16397
20 LET E = PEEK 16400+256*PEEK 16401
30 FOR Q = 1 TO 256
40 IF (Q/33) = INT(Q/33) THEN GOTO 60
50 POKE S + Q, 10
60 NEXT Q
```

The purpose of line 40 is to protect the Newline characters at the end of each line in the display file. Change the value to be Poked into memory in line 50 and the resulting screen display will ed different.


Sinclair ZX81 Personal Comp
the heart of a system
that grows with you.
1980 saw a genuine breakthrough the Sinclair ZX80 world's first complete personal computer for under £100. Not surprisingly over 50 000 were sold.
In March 1981 the Sinclair lead increased dramatically. For just £69.95 the Sinclair ZX81 offers even more advanced facilities at an even lower price. Initially even we were surprised by the demand – over 50 000 in the first 3 months!
Today the Sinclair ZX81 is the heart of a computer system. You can add 16 times more memory with the ZX RAM pack. The ZX Printer offers an unbeatable combination of performance and price. And the ZX Software library is growing every day.
Lower price higher capability
With the ZX81 it's still very simple to teach yourself computing but the ZX81 packs even greater working capability than the ZX80.
It uses the same micro-processor but incorporates a new more powerful 8K BASIC ROM the trained intelligence of the computer. This chip works in decimals handles logs and trig allows you to plot graphs and builds up animated displays.
And the ZX81 incorporates other operational refinements – the facility to load and save named programs on cassette for example and to drive the new ZX Printer.
Kit:
£49.95
Higher specification, lower price how's it done?
Quite simply by design. The ZX80 reduced the chips in a working computer from 40 or so to 21. The ZX81 reduces the 21 to 4!
The secret lies in a totally new master chip. Designed by Sinclair and custom built in Britain this unique chip replaces 18 chips from the ZX80!
New improved specification
● Z80A micro-processor new faster version of the famous Z80 chip widely recognised as the best ever made
● Unique one touch key word entry the ZX81 eliminates a great deal of tiresome typing. Key words (RUN LIST PRINT etc) have their own single key entry
● Unique syntax check and report codes identify programming errors immediately
● Full range of mathematical and scientific functions accurate to eight decimal places
● Graph drawing and animated display facilities
● Multi dimensional string and numerical arrays
● Up to 26 FOR/NEXT loops
● Randomise function – useful for games as well as serious applications
● Cassette LOAD and SAVE with named programs
● 1K byte RAM expandable to 16K bytes with Sinclair RAM pack
● Able to drive the new Sinclair printer
● Advanced 4 chip design micro-processor ROM RAM plus master chip – unique custom built chip replacing 18 ZX80 chips
ZX81
Built:
£69.95
Kit or built – it's up to you!
You'll be surprised how easy the ZX81 kit is to build just four chips to assemble (plus of course the other discrete components) a few hours work with a fine-tipped soldering iron. And you may already have a suitable mains adaptor – 700 mA at 9 V DC nominal unregulated (supplied with built version).
Kit and built versions come complete with all leads to connect to your TV (colour or black and white) and cassette recorder.
ZX81 BASIC
New BASIC manual
14



Available now- the ZX Printer for only £59.95
Designed exclusively for use with the ZX81 (and ZX80 with 8K BASIC ROM) the printer offers full alpha numerics and highly sophisticated graphics.
A special feature is COPY which prints out exactly what is on the whole TV screen without the need for further intructions.
At last you can have a hard copy of your program listings particularly
useful when writing or editing programs.
And of course you can print out your results for permanent records or sending to a friend.
Printing speed is 50 characters per second with 32 characters per line and 9 lines per vertical inch.
The ZX Printer connects to the rear of your computer – using a stackable connector so you can plug in a RAM pack as well. A roll of paper (65 ft long x 4 in wide) is supplied along with full instructions.
How to order your ZX81
BY PHONE – Access, Barclaycard or Trustcard holders can call 01 200 0200 for personal attention 24 hours a day every day.
BY FREEPOST – use the no-stamp needed coupon below. You can pay
by cheque, postal order, Access, Barclaycard or Trustcard.
EITHER WAY – please allow up to 28 days for delivery. And there's a 14-day money-back option. We want you to be satisfied beyond doubt – and we have no doubt that you will be.
16K-byte RAM pack for massive add-on memory.
To: Sinclair Research Ltd, FREEPOST, Camberley, Surrey GU15 3BR
Order
Qty
Item
Code
Item price £
Total £
Post and Packing
TOTAL £
FREEPOST – no stamp needed. Offer applies to UK only.
sinclair
ZX81
6 Kings Parade, Cambridge, Cambs., CB2 1SN.
Tel: (0276) 66104 & 21282.

Open Forum

OPEN FORUM is for you to publish your programs and ideas. It is important that you make sure your programs are bug free before you send them in. We cannot test all of them.

As far as possible always send us a computer printed listing made immediately after you have run the program.

Each program should include a description of what the program does, how to use it and how the program has been written.

Your contributions should be sent to Open Forum, *Popular Computing Weekly*, Hobhouse Court, 19 Whitcomb Street, London WC2H 7HF with a stamped addressed envelope.

We will pay £10 for each contribution published.

Anti-mind ***ZX81***

It's about time that your ZX81 started working for its living, and this is just the program to make it do so. Instead of forcing you to think, and play Mastermind against it, this program plays Mastermind against you; that is, it tries to guess your four-digit code.

Your code must be made from the digits 1 to 9. Once the computer has made a guess, enter its score as a single string (such as 21 or 03), the first digit being the number of blacks (correct digit in correct place), the second being the 'whites' (correct digit in the wrong position).

The program is rather slow and can take up to four minutes to make a decision, but the average response time is about one minute. The ZX81 can take up to nine guesses to crack your code, but is unlikely to need more than seven guesses.

Always run it in FAST.

Antimind

by Graham Charlton

```
[illegible]
 [illegible] "BLACK WHITE"
[illegible]
 30 LET T$="123456123456" [illegible] INT (RND*6)-1 TO [illegible]
 40 LET [illegible]=1
 50 LET [illegible]=1
 60 LET [illegible]=1
 70 F[illegible] TO 4
 80 F[illegible] TO 6
 90 F[illegible] TO 8
100 LET [illegible]
120 LET A[illegible]-Y
130 LET H[illegible]
140 LET H[illegible]
150 IF [illegible] THEN GOTO 30
160 F[illegible] TO [illegible]
170 LET [illegible]
180 LET [illegible]
190 LET L=H[illegible]
200 LET M=H[illegible]
210 LET N=H[illegible]
220 F[illegible] TO 4
230 LET [illegible]
240 LET E=E-[illegible]
250 NEXT F
260 LET E=E+10[illegible]
270 IF E[illegible] THEN GOTO [illegible]
280 PRINT T
290 PRINT AT [illegible]
300 [illegible]
310 PRINT [illegible] H$ [illegible] A$ [illegible]
320 LET H[illegible] VAL H$
330 LET [illegible]=0
340 LET [illegible]+1
350 IF [illegible] THEN PRINT [illegible]
360 IF VAL H$=40 THEN PRINT "I GUESSED YOUR CODE IN [illegible] TRIES" [illegible]
370 IF [illegible] THEN LET [illegible]
380 IF [illegible] H$=10 THEN GOTO 440
390 IF VAL A$=20 THEN GOTO 440
400 IF VAL A$=40 THEN GOTO 420
410 NEXT W
420 NEXT Y
430 NEXT [illegible]
440 NEXT [illegible]
450 PRINT "[illegible] THINK YOU CHEATED [illegible]"
```

Squash ***BBC***

In this version of *Squash* written in Mode 7 for the BBC Microcomputer, you use the Z and M keys to move the slide at the bottom of the screen right and left respectively.

The program keeps track of how long you keep the ball in flight and at the end gives you a score based on this time. Pressing any key at the end

of the game will give you a new game. There is a highest score feature.

You can make the game easier by changing the zero after INKEY$ in line 140 to a 5 or a 10, and add a delay loop between lines 170 and 180. You may also wish to change the sound the ball makes when it bounces. Simply change the third figure after the word SOUND in lines 650 and 660. Try 120 + RND(130) or 240 + RND(14) for two different effects.

Subchase *ZX81*

The main problem in using the ZX81 for games is the slowness of the Basic compared to that on more expensive machines.

The slowness is due largely to the way the display is handled. But when designing a program, you can give the illusion of increased speed, where in fact the speed remains constant.

In *Subchase* the illusion of speed is given by altering the relevant display as the value is calculated. Thus the display is continually changing to give the effect of increased speed. If the display were updated after all the values had been calculated, the response would be slow.

Subchase also demonstrates how to use a graphic display employing PLOT to help increase speed.

Now let's look at the game in detail. Imagine yourself at the helm of a ship; your mission is to catch and destroy an enemy submarine. The program listed here is a real time program that simulates the action.

You have control of throttle and steering; the throttle is controlled by the numbers 0 to 9, and the steering by S, L and R, and depth charges are launched with key D.

L gives a 1 degree offset of course to port indicated by −1, and R gives a 1 degree offset of course to starboard indicated by +1. This course change is in force till the S key is pressed; this gives a course offset of zero. The offset to port or starboard can be set to − + 10 degrees by using L/R with the shift key.

The display shows your speed and heading — the small marker above the scale. The target's speed, heading, bearing and distance are shown by the marker below the scale.

The throttle is such that a value of 5 keeps your speed constant; a value other than 5 slows you down or speeds you up at a rate of Throttle-5. The target's speed and heading remain constant until the sub is damaged; then its speed decreases by a set factor and its heading changes randomly.

The program was built on a 4K system and left little space for expansion, so the actual attack routine was left with room for improvement. However, the program gives the user a starting point as far as the trigonometry of such an operation is concerned, and with a 16K RAM pack the program can be advanced to a more accurate simulation.

The key inputs are done using the INKEY$ function, so the input will at times be slow.

The program uses the built-in timer of the ZX81 to decide the time lapse to update the various displays. The distance is only displayed when the value is less than 6300 feet, and when less than 630 feet the scale is enlarged.

When you are within 200 feet of the submarine you will be informed that the target is in range. Within 100 feet the sub will be hit; within 200 feet the sub will be damaged.

Happy hunting.

See page 18 for the full listing of *Subchase*

Squash
by Tim Hartnell

```
10 REM * SQUASH *
20 REM move bat using the Z and M keys
30 REM (C Hartnell 1982
40 REM FROM Let Your BBC Micro
50 REM Teach You to Program
60 MODE 7
70 HIGH=0:REM
80 PROCset up
90 REM****************
100 REPEAT
110 PRINT TAB(8,[illegible]) CHR$ 129+RND 5 [illegible] score is [illegible]
  HIGHSCORE
120 PRINT TAB [illegible] CHR$(RND 5)+128 "You have lost the ball
  [illegible]
130 PRINT CHR$ RND 5) 129 [illegible] TIME DIV 10 [illegible] seconds
140 A$=INKEY$ 0
150 REM****************
160 *FX 15 1
170 IF A$= [illegible] OR A$= M [illegible] PROCmove bat
180 PROCmove ball
190 PRINT TAB 0+11 19 [illegible]
200 UNTIL end
210 IF TIME DIV 100>HIGHSCORE HIGHSCORE=TIME DIV 100
220 TIME=0
230 REPEAT UNTIL TIME>[illegible]
240 *FX 15 1
250 FOR I=0 T[illegible]
260 SOUND 1,-15 [illegible] RND [illegible] 100 25
270 NEXT I
280 TIME=0
290 REPEAT UNTIL TIME>150
300 *FX 15 0
310 *FX 3[illegible]
320 GOTO 70
330 REM****************
340 DEF PROCset up
350 LOCAL
360 PRINT TAB 10 10 [illegible]
370 [illegible] TAB 10 10
380 PRINT TAB(10 7 0 [illegible] TAB(30 T+10 CHR$[illegible]
390 NEXT I
400 BAT$=" —
410 NOBAT$=
420 X=1
430 Y=1
440 L=1
450 M=1
460 TIME=0
470 B=10
480 PRINT TAB 11+B 19 BAT$
490 *FX 11 1
500 *FX 12 1
510 VDU 23 8202 0 0 0
520 ENDPROC
530 REM****************
540 DEF PROCmove bat
550 IF A$= M AND B=18 THEN ENDPROC
560 IF A$= [illegible] AND B=0 THEN ENDPROC
570 PRINT TAB 11+B 19 NOBAT$
580 IF A$= M THEN B=B+1
590 IF A$="Z" THEN B=B-1
600 PRINT TAB 11+B 19 BAT$
610 ENDPROC
620 REM****************
630 DEF PROCmove ball
640 PRINT TAB 11+X 11+Y " "
650 IF L+X<10 OR L+X>20 THEN L=-L SOUND 1 [illegible] 15 [illegible]
660 IF M+Y<0 OR M+Y>[illegible] THEN M=-M SOUND 1 [illegible] 15 [illegible]
670 X=X+L
680 Y=Y+M
690 PRINT TAB(11+X 11+Y) "O"
700 IF Y=8 AND [illegible] THEN end=TRUE ELSE end=FALSE
710 P=TIME
720 ENDPROC
730 REM new game
740 PRINT [illegible] CHR$(RND 5)+128 "Press any key for a new
  game
750 A$=GET$
760 CLS
770 GOTO 80
```



Utility No 1
by Ken Mahogany

NB – On this listing the hash sign (#) signifies pounds (£)

Sample run

Subchase
by John Sylvester

Utility programs — ZX81

Here are two utility programs for the ZX81 which demonstrate convincingly that the power of the Sinclair computer is not limited to playing games.

The first program produces a straight line depreciation table as can be seen from the sample run. Note the effect of line 140 which stops the computer from working in fractions of a pound less than 1p. Without this feature the output could contain absurd data such as £123.87654.

The second program — Mean Standard Deviation and Variance — uses data entered by the user to produce the three outputs mentioned in the title. You enter the items of data one by one followed by the frequency and then enter $ when you wish the computer to start processing.

Layout planner — ZX81

Moving house? Reorganising the office? This program can help you with your planning.

Key in the dimensions of a piece of furniture in either feet and inches or millimetres and the ZX81 will print a rectangular cut-out of the appropriate dimensions. The scale is such that the full width of the paper — 90 mm — represents approximately nine feet or three metres so you can draw a plan of your room accordingly.

The program allows you to key in a description of each item which must of course be short enough to fit into the size of the rectangle. Note that if you are using imperial measurements the program will accept a measurement such as 38 inches but only if zero is entered against feet.

The appearance of the program can be improved by using inverse video for the messages. Store requirement is about 2.5K.

See page 20 for the full listing of *Layout planner*.

Cricket — ZX81

This is a simple reaction test game: you have to press the correct key within the time allowed, or else you are bowled out.

The game can be made easier or harder by altering the length of the pause at line 140.

See page 20 for the full listing of *Cricket*.

Utility No. 2
by Ken Mahogany

```
10 REM MEAN  STANDARD
20 REM DEVIATION
30 REM VARIANCE
40 REM  L  MAHOGANY 1982
44 SCROLL
45 PRINT "ENTER $ TO TERMINATE ENTRIES"
50 DIM A(50)
60 DIM B(50)
70 LET Z=0
80 LET Z=Z+1
90 SCROLL
95 SCROLL
100 PRINT "ENTER ITEM ";Z
110 INPUT X$
120 IF X$="$" THEN GOTO 1000
130 LET NUMBER=VAL (X$)
140 LET A(Z)=NUMBER
160 PRINT NUMBER
170 SCROLL
180 PRINT "ENTER FREQUENCY ",
190 INPUT FREQ
200 LET B(Z)=FREQ
210 PRINT FREQ
220 GOTO 80
1000 LET P=0
1010 LET O=0
1020 LET L=0
1030 FOR W=1 TO 50
1040 LET P=P+B(W)
1050 LET O=O+B(W)*A(W)
1060 LET L=L+B(W)*A(W)*A(W)
1070 NEXT W
1080 SCROLL
1090 SCROLL
1100 SCROLL
1110 PRINT "MEAN IS ";O/P
1120 SCROLL
1130 PRINT "VARIANCE IS ";(L*P-O*O)/(P*P)
1140 SCROLL
1150 PRINT "STANDARD DEVIATION IS ";SQR ((L*P-O*O)/(P*P))
```

Sample run

```
ENTER $ TO TERMINATE ENTRIES

ENTER ITEM 1    23
ENTER FREQUENCY 4

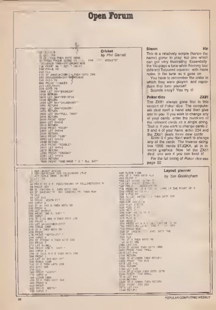
ENTER ITEM 2    24
ENTER FREQUENCY 4

ENTER ITEM 3    25
ENTER FREQUENCY 4

ENTER ITEM 4

MEAN IS 24
VARIANCE IS 0.66666675
STANDARD DEVIATION IS 0.81649553
```

Open Forum



Cricket

by Phil Garratt

```
130 PRINT AT 0,0;" HIT ";B$(I)
140 PAUSE 50
160 CLS
170 IF INKEY$=CHR$(I) THEN GOTO 200
180 GOSUB 2000+INT(RND*5)*20
190 GOTO 70
200 PRINT "TIMED"
210 LET V=V+1
220 GOTO 70
1000 LET B$="BOUNCER"
1010 RETURN
1020 LET B$="OFF-SPIN"
1030 RETURN
1040 LET B$="IN-WINDER"
1050 RETURN
1060 LET B$="YORKER"
1070 RETURN
1080 LET B$="FULL TOSS"
1090 RETURN
2000 PRINT "SIX"
2005 LET X=X+6
2010 RETURN
2020 PRINT "FOUR"
2025 LET X=X+4
2030 RETURN
2040 PRINT "TWO"
2045 LET X=X+2
2050 RETURN
2060 PRINT "SINGLE"
2065 LET X=X+1
2070 RETURN
2100 PRINT "CAUGHT"
2105 LET V=V+1
2110 RETURN
9000 PRINT "YOU MADE ";X;" ALL OUT"
```

Simon — Vic

This is a relatively simple (hence the name) game to play, but one which can get very frustrating. Essentially the Vic plays a tune while flashing four different coloured squares, with more notes in the tune as it goes on.

You have to remember the order in which they were played and reproduce that tune yourself.

Sounds easy? You try it!

Poker dice — ZX81

The ZX81 always goes first in this version of Poker dice. The computer will deal itself a hand and then give one to you. If you wish to change any of your cards, enter the numbers of the relevant cards as a single string. That is, if you want to change cards 2, 3 and 4 of your hand, enter 234 and the ZX81 deals three new cards.

Enter 0 if you don't want to change any of the cards. The inverse string line 1000 reads 9TJQKA, all in inverse graphics. Now let the ZX81 deal and see if you can beat it!

For the full listing of *Poker dice* see page 22

Layout planner

by Tim Goldingham

Open Forum

Simon
by Dave Middleton

```
[illegible]
1000 PRINT [illegible] LEFT$ [illegible]
[illegible]
PRINT [illegible] LEFT$ [illegible]
[illegible]
RETURN
[illegible] CT=205 GOSUB1000 [illegible]
RETURN
[illegible] GOSUB1000 [illegible]
RETURN
[illegible] CT=225 [illegible]
RETURN
[illegible] PRINT "DIFFICULTY LEVEL [illegible]
[illegible] PRINT [illegible] RETURN
[illegible]
RETURN
[illegible]
PRINT [illegible]
[illegible] THEN [illegible]
[illegible] RETURN
```

Open Forum

Poker dice
by Tim Hartnell



```
10 GOTO 1000
20 LET A(A)= INT (RND*6)+1
30 LET B$(2*A) A$(A(A))
40 RETURN
50 FOR A=1 TO 5
60 GOSUB 20
70 LET B(A(A))=B(A(A))+1
80 NEXT A
90 PRINT TAB 10 B$
100 PRINT
110 FOR G=1 TO 2
120 GOSUB 500
130 PRINT TAB 10;B$
140 PRINT
150 NEXT G
160 PRINT TAB 12 "PLAYER"
170 PRINT
180 FOR C=1 TO 5
190 LET C$(2*C)=A$(INT (RND*6)+1)
200 NEXT C
210 PRINT TAB 10 C$
220 PRINT
230 FOR G=1 TO 2
240 INPUT D$
250 IF D$ "0" THEN GOTO 300
260 FOR C=1 TO LEN D$
270 LET C$ *VAL D$    A$ NT RND*6)+1)
280 NEXT
290 PRINT TAB 13
300 PRINT
310 NEXT G
320 FOR A 1 TO 200
330 NEXT A
340 CLS
350 RUN
500 DIM R(6)
510 FOR B=1 TO 6
520 IF B(B)>2 THEN LET R(B) 1
530 NEXT B
540 FOR A 1 TO 5
550 IF R(A(A)) 1 THEN GOSUB 20
560 LET B(A(A))=B(A(A))+1
570 NEXT A
580 RETURN
1000 LET A$ "STUQKA"
1010 DIM B$(10)
1020 DIM A(5)
1030 DIM B(6)
1040 DIM C$(10)
1050 PRINT TAB 13 "ZX81"
1060 PRINT
1070 GOTO 50
```

Peek & poke

Peek your technical queries to Graham Charlton He will poke back an answer

HOW TO RANDOMIZE COLOUR ON A VIC?

Charles Secker of Willow Road Cannock Staffs writes

Q I have written an adventure game for my Vic 20 and would like to know if it is possible to add a routine which will automatically change print statements in a random colour so that if the same message comes up more than once in a game it will not necessarily appear in the same colour each time. Is there a routine that can do this please?

A It can be done fairly easily by assigning a string, say A$, to all the colour codes at the start of the game. Then you need a subroutine which assigns B$ to one element of A$. The main line of the subroutine should be

```
B$=MID$(A$,INT(RND
(1)*7)+1,1)
```

Then, when you return to the main program use a line beginning

```
PRINT B$        "all your words
here"
```

This should produce the effect you want.

BUBBLE BUBBLE RAM IS NO TROUBLE

John McPherson of Lands-down Grove Swansea writes

Q I have just built myself a 16k memory unit and have become very interested in how computer memory works. I have some idea but would like to know more, especially about bubble memories which I saw demonstrated at the Challenge of the Chip exhibition at the Science Museum last year. Could you suggest a source of information on computer memory for me?

A There are a number of good (and pretty complicated) books which discuss computer memory. A good general introduction to the subject is Walter H Buchsbaum's book *Personal Computers Handbook* published by Howard W Sams and Co.

Other books which treat the subject in more detail include:
- *Computer Storage Systems and Technology* by R Matick from John Wiley and Sons
- *Memory Design Microcomputers to Mainframes* in the Electronic Book Series from McGraw Hill
- *Microprocessor Applications Manual* via Motorola Semiconductor Products from McGraw Hill
- *Introduction to Microprocessors* by L A Leventhal from Prentice Hall

WHERE HAVE ALL THE ZX USERS GONE?

Martin [illegible] of [illegible] Gardens [illegible] writes

Q I want to get in touch with other ZX81 users in my own area. Can you put me in touch with them please?

A There is a local ZX81 Users Group in Orpington Kent organised by Roger Pyall at 23 Arundel Drive. You can phone him on (66) 20281.

If you would prefer to start your own group we suggest you place an advertisement in a local paper saying something like "Interested in the ZX81? So am I. Please get in touch so we can share programs", then put your name and address or phone number.

I think you'll be pleasantly surprised at the response you get.

BBC KEEP QUIET ON MICRO SOUND FX

Graham Blackm[illegible] of Bridge Lane St Leonards writes

Q I have a BBC Microcomputer and find the instructions regarding the sound in the manual absolutely useless. All it says is "The syntax for this statement was not available in time for this provisional User Guide". Could you please tell me how they work?

A The BBC estimate they will have to send out some 30,000 copies of the new User Guide when it is complete. From the proofs of the new guide, and from my own experiments on the computer, it is clear that the SOUND command is very easy to use and if you avoid ENVELOPE the first few times you use the computer's sound facility you'll find SOUND easy to master.

The word SOUND has to be followed by four parameters. The first (0 to 3) chooses the sound channel, the second (-1 to -15) sets the volume, the third chooses the pitch of the note (1 to 254) and the fourth controls its duration.

A simple one line program which puts all the SOUND output through its paces indefinitely producing a sort of weird electronic music is:

```
10 SOUND RND(4)-1,-RND
   (15),RND(254),RND(20)
20 RUN
```

The sound command is very versatile as can be seen from the following program which produces an effect you may well wish to incorporate into a program. It is only one sample of the infinite range of sounds available.

```
10 SOUND 0,-15,100,6
20 SOUND 1,-15,100,5
30 SOUND 2,-15,140,4
40 SOUND 3,-15,100,3
```

THE HILLS ARE ALIVE WITH THE SOUND OF ...

Paul Jacobson of Sandy Hill Road Birmingham writes

Q I have heard that it is possible to play music on a ZX81. I can't see how this could be done short of somehow modifying the horrible sound that the computer makes when it is SAVEing. Could you please tell me if it is possible to use the ZX81 for music and if so how it would be done?

A It is possible to produce music of sorts from the ZX81 although the sound is not good enough to allow the computer to be used as a true musical instrument.

You need a machine code routine and a good one is given in the book *Mastering Machine Code on Your ZX81*. Author Toni Baker makes some pretty extravagant claims for this routine, saying for example "... one big advantage machine code does have over Basic is you can ..." and the program is in machine code, not Basic.

If you don't want to go to the trouble of working in machine code a music of sorts (of sorts) can be written in Basic. Try the following and see what it produces from your TV's speaker.

```
10 REM *MUSIC
        (C)
    MAHOGANY 1982
20 FOR Z=1 TO RND*80
30 SLOW
40 FAST
50 NEXT Z
60 FOR Z=1 TO RND*80
70 SLOW
80 FAST
90 NEXT Z
100 FOR Z=1 TO RND*80
110 SLOW
120 FAST
130 NEXT Z
140 FOR Z=1 TO RND*80
150 SLOW
160 FAST
170 NEXT Z
180 PAUSE RND*10
190 RUN
```

I WANT TO BUILD UP MY ATOM

Tony Asherwood of Hilltop Lane Paignton Devon writes

Q I have an Acorn Atom which I bought unfortunately just before the news that Acorn would build the BBC Micro was announced. Is it possible to upgrade my Atom to BBC specifications through some EPROM or something similar? What other EPROMs are available for the Atom?

A Acorn originally promised that EPROMs to bring the Atom some of the way towards the BBC Micro would be available in early January but they do not seem to have appeared.

When they do it will give your Atom the facility of working on a generous subset of BBC Basic.

There is a 4k ROM version of Viewcalc available for the Atom which says Acorn can cope with anything from a household budget to company accounts.

> Send your questions to Peek & poke, *Popular Computing Weekly*, Hobhouse Court, 19 Whitcomb Street, London WC2 7HF

It's easy once you can speak the language

***David Kingsbury** puts in a good word for assembler or low-level languages*

MANY TEACHERS of computer studies believe assembly language programming is destructive because it can confuse pupils. They say it does not allow an easy appreciation of the structure and flow of good programming.

As with all good listening sense there is just enough truth in this notion to allow someone to extend the argument beyond recognition. Many of those who put down assembly languages or assemblers use and promote Basic or another high level language such as Pascal, Comal or Forth.

Here are some harsh facts.

There is no best computer language because there is no best job for a computer to do. But some languages are very successful for certain tasks.

Basic is a good language for beginners to learn about computing, but it is a dreadful distortion of Basic to use it for word processing. It is also impossible to write high speed programs to produce pictures or graphic displays in Basic.

Fortran, the grand-daddy of high-level languages is a good language for scientific applications.

You will get faster

Forth and languages such as Stoic are arguably the most general purpose languages currently available, fast, versatile, suitable both for controlling aluminium smelters and for handling text manipulation. They are not easy, however, to read at the more primitive levels of a program's development.

Assemblers are low level computer languages in which one instruction by the programmer produces one instruction in machine code for the computer to obey. Consequently it is difficult to write programs in assembler quickly if your program is still in the early stages of its development.

But if you really understand the purpose of your program and the steps you must carry out to achieve the ultimate aim, programming in assembler will get faster and faster as you combine subroutines and modules that you have already written.

Let's look at how programs can be constructed using standard assembler instructions. Suppose you came across the following program:

```
1           BUTTERCUP ZERO
2           COWSLIP SPACE
3   TRIFLE  DOGWORT (SCREENTOP) Y
4           ROBCHIP
5           HARESFOOT LINELENGTH
6           MEADOWSAFFRON TRIFLE
```

You might reasonably regard this with some suspicion. But it is just a program written in assembly language to clear the top line of the screen in a computer using a 6502 central processor unit.

Any names would do

Look at the program again when it is written like this:

```
1           LDY ZERO
```

Load register Y with zero

```
2           LDA SPACE
```

Load the accumulator with a space character

```
3   TRIFLE  STA (SCREENTOP), Y
```

Store the value in the accumulator in the memory location pointed to by adding the variable SCREENTOP to the value in register Y

```
4           INY
```

Add one to the value in register Y

```
5           CPY LINELENGTH
```

Compare the value in Y with the value in the variable LINELENGTH

```
6           BNE TRIFLE
```

Branch to the label TRIFLE if Y is not equal to LINELENGTH

The point of the herbal names in the first version is only to point out that an assembler could use any names or mnemonics to describe the same action. For example MEADOW SAFFRON may be BNE or anything else in source code, so long as it stores the 6502 machine code instruction D0 when the program is assembled to object code.

Early assemblers for microcomputers sometimes used non-standard mnemonics but you should now be able to buy an assembly language program for any microcomputer that uses standard names for each instruction strictly in accordance with the central processor unit's manufacturers specifications.

A lot of talk goes on about structured languages. Or how isn't it good that PASCAL is structured and isn't it bad that BASIC just works without structure.

Programming with a clear idea of what you want to achieve is most likely to be successful and if you break the task down into smaller jobs and bits that you will do many times then you acquire structure without really noticing.

Nevertheless, a well-designed language can make you think about a problem in ways that will contribute to its solution. The instructions

```
DO      UNTIL
REPEAT  WHILE
IF      THEN   ELSE
```

are generally held to be structured in the sense that they define a module in the program with a known exit.

Look for structure

Assembly language programs can have as much or more structure than higher-level programs. Look at this short program for example:

```
1   WAITCR  PHP
2           PHA
3   W1      JSR GETCHAR
4           LDA CHARACTER
5           CMP CARRIAGERETURN
6           BNE W1
7           PLA
8           PLP
9           RTS
```

That is a DO ... UNTIL construction in that the program carries out the instructions in the loop formed by lines 3 to 6 until the key pressed by the operator is the Carriage return or newline key.

If you are approaching this subject for the first time then some of the terms will be unfamiliar. The easiest way to overcome the problem is to acquire a well written book and read it while using your computer.

For people with a 6502 based machine there is nothing better than *6502 Software Design* by Leo J Scanlon (Howard W Sams & Co Inc ISBN 0-672 21656-6).

Other books are available for Z80 based computers. For example the one by Adam Osborne is quite adequate.

```
100 REM
200 REM  INITIALIZE
300 REM
400 V1=36874   V2=36875   V3=36876   V4=36877
500 REM  SET VOLUME
600 POKE V4
700 REM
800 REM INPUT NOTES
900 REM  KEY OF C
1000 REM
1100 DIM A(15)
1200 FOR I = 1 TO 15  READ A(I)   NEXT I
1300 DATA 195 201 207 209 215 219 223 225 228 231
1400 REM
1500 REM INPUT USER DEFINED TUNE
1600 REM
1700 PRINT "HOW MANY NOTES "  INPUT N
1800 DIM NN(N,3)
1900 PRINT "ENTER TUNE" PRINT  VOICE1 VOICE2 VOICE3 RETURN
2000 FOR I=1 TO N   INPUT NN(I,1)  NN(I,2)  NN(I,3)  NEXT I
2100 REM
2200 REM NOW PLAY TUNE
2300 REM
2400 FOR I=1 TO N
2500 POKE V1 A(NN(I,1)) POKE V2 A(NN(I,2))  POKE V3 A(NN(I,3))
2600 FOR J=1 TO 100   NEXT J
2700 POKE V1 0 POKE V2 0   POKE V3 0
2800 NEXT I
2900 PRINT  PLAY TUNE AGAIN   INPUT W$   IF W$="Y"  THEN GOTO 2400
```

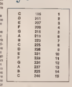
Stick this in your Vic and play a tuna

The Vic 20 microcomputer from Commodore is an ideal first buy for any user interested in microcomputer music. Although you won't quite be competing with Depeché Mode from the day your Vic arrives the sounds that the computer makes possible are to say the least interesting

I found that after a while the limitations of the Vic started to become annoying but for beginners it is fine The graphics are good too

The listing given here provides the kernel of a music making program which will give you some ideas to work from You can make it more sophisticated — especially with regard to the input routines — in a number of ways A smart screen display wouldn't go amiss either I left those so you can slot them in to suit your own tastes

To keep things simple the notes stored in the array A are two octaves worth of the key of C If your ears are very good you will notice that the notes are not exactly perfect but for now they will pass muster Figure 1 shows the notes together with the values that need to be poked into the Vic's voice registers to obtain those notes the third column shows the positions of the notes in the array



C	195	1
D	201	2
E	207	3
F	209	4
G	215	5
A	219	6
B	223	7
C	225	8
D	228	9
E	231	10
F	232	11
G	235	12
A	237	13
B	225	14
C	240	15

Figure 2 shows a musical stave The names of the notes are marked on the stave in their correct positions To the left of the stave is a list of the numbers relating to each of the notes and to the right is the position of the note in array A. Enter a zero to output a silence from any one of the voices at any point in the tune

The program is pretty straightforward When it is run the question How many notes? appears On an unexpanded Vic you will be able to store around 130 notes on a Vic with the 3K super-extender this becomes about 260 notes After that you have to input the notes giving the numbers in figures 1 and 2 — voice 1 first followed by a comma then voice 2 comma voice 3 return The tune will then play

If the volume on your tv is set too loud you may jump out of your skin But you are now making microcomputer music.

Try buying the sheet music of your favourite songs and enter them but remember that the timings are not programmed yet **Sam Blythe**

Competitions

1 Here's how you can win a ZX81 16K RAM

Each week we plan to give away a valuable piece of hardware as a prize for the best program written by you the readers on a theme set by us.

The program must be no longer than 100 lines, and be suitable for the Vic 20, the ZX81 or the BBC Micro.

Your entry should be accompanied by an explanation of the function and use of the program and also of how you went about writing it.

The winner will be the reader who in the opinion of the Editor of *Popular Computing Weekly* has sent in the most original and inventive program.

In all cases the decision is final.

From time to time we will also publish programs which do not win the competition but which we think will be interesting to other readers.

If that happens we will pay the usual rate for contributions to the magazine.

This week's fantastic prize is a ZX81 16K RAM pack and the theme for the competition is **Treasure Island**.

Send your program to The Editor, *Popular Computing Weekly*, 16 Whitcomb Street, London WC1.

Please mark your entry Hardware Competition 2 — and remember it must reach us by first post May 10, 1982.

2 Complete the crossword and win £10

ACROSS

1 Dev[illegible]
6 Records with [illegible]
9 Shows [illegible]
Over [illegible]
13 Spiked [illegible]
14 Dad [illegible]
Shel[illegible]
18 [illegible]
20 [illegible]
21 The [illegible]

DOWN

1 Obscure [illegible]
2 You're [illegible]
3 Permitted [illegible]
4 Representative of a [illegible]
5 Operator using the [illegible] the best part (3)
6 Four-bit representation [illegible]
7 Ruler giving direction [illegible]
8 Cleans floor spaces
10 British [illegible] (5)
16 Texas Instruments up in 150 [illegible] (3,3)
17 Quickly re-advertise [illegible]
19 Do the wrong [illegible]

3 Solve the puzzle and win a gift voucher!

The staff of the Neverworks Telephone Company played an April Fool's day joke on the foreman.

Some of the apprentices made a special joke dial that they slipped into a batch of phone dials which had been delivered. As you can see the figures were all jumbled up.

0 7 1 5 4 9 8 2 3 6

The foreman was most disconcerted as he assumed that the whole batch was faulty. He got so angry that he threw the joke dial against the wall.

The apprentices found that the dial had broken into three pieces and that the number shown by the digits on one of the pieces was the result that would be obtained by multiplying the numbers produced by the digits (or digit) on both of the other two pieces.

In each case the number was that obtained by reading round in a clockwise direction.

What were the numbers?

Readers can enter any or all of our competitions but please use a separate envelope for each as this helps our judges.

The winner for the crossword and the winner of the puzzle in each case will be the first name out of the hat.

Closing date for both the crossword and the puzzle is the Monday three weeks after the cover date.

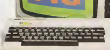
Please mark your envelope "CROSSWORD" or "PUZZLE".

CITIZEN PAIN

BY DAVID IRELAND and JAMES MACDONALD







"Give me one good reason why I should choose a VIC 20 home computer."

1. VIC is outstanding value for money. No other colour home computer can give so much for under £200.
2. Total standard memory 25K made up of 20K ROM and 5K RAM.
3. Fully expandable to 27K user RAM.
4. Microsoft Basic interpreter as standard.
5. Accessible machine language via plug-in cartridges.
6. Connects direct to monitor or standard television.
7. Full size typewriter style keyboard.
8. Full colour and sound.
9. All colours easily accessible.
10. 62 predefined graphic characters direct from the keyboard.
11. Full set of upper and lower case characters.
12. 256 displayable characters direct from the keyboard.
13. High resolution graphics capability via plug-in cartridges.
14. Programmable function keys can be used with plug-in cartridges.
15. Automatic repeat on cursor function keys.
16. User definable input/output port.
17. Machine bus port for memory expansion and ROM software.
18. Standard interfaces for hardware peripherals.
19. VIC 20 is truly expandable into a highly sophisticated computer system with a comprehensive list of accessories (see panel below).
20. Full range of software for home, education, business and entertainment on disk, cassette and cartridge.
21. Books manuals and learning aids from Teach Yourself Basic to the VIC programmers' reference guide (a must for advanced programmers).
22. Full support for VIC owners – their own magazine 'VIC Computing' as well as a national network of VIC user groups.
23. National dealer network providing full service and support to VIC owners.
24. Expertise and experience – Commodore are world leaders in microcomputer and silicon chip technology.
25. Commodore is the leading supplier of micro computers in the UK to business, schools, industry and the home.
26. VIC 20 is the best-selling colour home computer in the UK.

How many reasons was it you wanted?

Accessories include:

- Cassette tape unit.
- Single drive 5¼" floppy disk unit (170 K bytes capacity).
- 80-column dot matrix printer.
- 3K, 8K and 16K RAM expansion cartridges.
- Programming aid and machine code monitor cartridge, programmers aid cartridge, high resolution graphics cartridge.
- Plug in expansion box for a full 32K, 40 column x 24 lines VIC including Prestel compatibility.
- RS 232C communication cartridge.
- Memory expansion board.
- IEEE488 interface cartridge.
- Joysticks, light pens, paddles and motor controllers.

commodore

VIC 20

The best home computer in the world.

FOR MORE INFORMATION ON THE VIC 20 TELEPHONE OR WRITE TO COMMODORE INFORMATION CENTRE 675 AJAX AVENUE SLOUGH, BERKSHIRE. TELEPHONE SLOUGH 79292